PORTAL

USPTO

Subscribe (Full Service) Register (Limited Service, Free) Login

Search: ● The ACM Digital Library ○ The Guide

+database +abstraction +layer +key +language SEARCH

Feedback Report a problem Satisfaction survey

Terms used **database** **abstraction** **layer** **key** **language** Found **3,093** of **169,866**

Sort results by: relevance

Display results: expanded form

Save results to a Binder

Search Tips

□ Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next

Best 200 shown

Relevance scale □ □ □ ■ ■

1 Article abstracts with full text online: π-ADL: an Architecture Description Language based on the higher-order typed π-calculus for specifying dynamic and mobile software architectures ■

Flavio Oquendo

May 2004 **ACM SIGSOFT Software Engineering Notes**, Volume 29 Issue 3

Publisher: ACM Press

Full text available: pdf(615.91 KB) Additional Information: full citation, abstract, references, citings

A key aspect of the design of any software system is its architecture. An architecture description, from a runtime perspective, should provide a formal specification of the architecture in terms of components and connectors and how they are composed together. Further, a dynamic or mobile architecture description must provide a specification of how the architecture of the software system can change at runtime. Enabling specification of dynamic and mobile architectures is a large challenge for an ...

Keywords: n-calculus, Architecture Description Languages, dynamic architectures, mobile architectures, specification languages

2 Fast detection of communication patterns in distributed executions ■

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research**

Publisher: IBM Press

Full text available: pdf(4.21 MB) Additional Information: full citation, abstract, references, index terms

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

3 Computing curricula 2001 ■

September 2001 **Journal on Educational Resources in Computing (JERIC)**

Publisher: ACM Press

Full text available: pdf(613.63 KB) Additional Information: full citation, references, citings, index terms html(2.78 KB)

4 Building a layered database for design automation
Robert V. Zara, David R. Henke
June 1985 **Proceedings of the 22nd ACM/IEEE conference on Design automation**
Publisher: ACM Press
Full text available: pdf(962.36 KB) Additional Information: full citation, abstract, references, citings, index terms

A layered approach is presented for the database of a distributed, interactive design automation system. Levels of abstraction are described from the point of view of the bottom-up designer. The controversy between the relational and network database formats is explored in the central abstraction: an object-oriented layer which attempts to select the advantages of each of these two formats while avoiding their respective disadvantages. This object-oriented approach treats each of ...

5 Modeling the storage architectures of commercial database systems
D. S. Batory
December 1985 **ACM Transactions on Database Systems (TODS)**, Volume 10 Issue 4
Publisher: ACM Press
Full text available: pdf(4.46 MB) Additional Information: full citation, abstract, references, citings, index terms, review

Modeling the storage structures of a DBMS is a prerequisite to understanding and optimizing database performance. Previously, such modeling was very difficult because the fundamental role of conceptual-to-internal mappings in DBMS implementations went unrecognized. In this paper we present a model of physical databases, called the transformation model, that makes conceptual-to-internal mappings explicit. By exposing such mappings, we show that it is possible to model the storage ...

6 The design of the E programming language
Joel E. Richardson, Michael J. Carey, Daniel T. Schuh
July 1993 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 15 Issue 3
Publisher: ACM Press
Full text available: pdf(2.78 MB) Additional Information: full citation, references, citings, index terms, review

Keywords: extensible database systems, persistent object management

7 Special issue on persistent object systems: Fibonacci: a programming language for object databases
Antonio Albano, Giorgio Ghelli, Renzo Orsini
July 1995 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 4 Issue 3
Publisher: Springer-Verlag New York, Inc.
Full text available: pdf(2.15 MB) Additional Information: full citation, abstract, references, citings

Fibonacci is an object-oriented database programming language characterized by static and strong typing, and by new mechanisms for modeling databases in terms of objects with roles, classes, and associations. A brief introduction to the language is provided to present those features, which are particularly suited to modeling complex databases. Examples of the use of Fibonacci are given with reference to the prototype implementation of the language.

Keywords: data models, database programming languages, objects with roles

8 On the decidability of semi-linearity for semi-algebraic sets and its implications for spatial databases (extended abstract)
Freddy Dumortier, Marc Gyssens, Luc Vandeurzen, Dirk Van Gucht
May 1997 **Proceedings of the sixteenth ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems**
Publisher: ACM Press
Full text available: pdf(1.69 MB) Additional Information: full citation, references, citings, index terms

9 Concepts for a database system compiler
D. S. Batory
March 1988 **Proceedings of the seventh ACM SIGACT-SIGMOD-SIGART symposium on Principles of database systems**
Publisher: ACM Press
Full text available: pdf(1.22 MB) Additional Information: full citation, abstract, references, citings

We propose a very simple formalism based on parameterized types and a rule-based algebra to explain the storage structures and algorithms of database management systems. Implementations of DBMSs are expressed as equations If all functions referenced in the equations have been implemented the software for a DBMS can be synthesized in minutes at little cost, in contrast to current methods where man-years of effort and hundreds of thousands of dollars are required. Our research aims to develop ...

10 The design and implementation of hierarchical software systems with reusable components
Don Batory, Sean O'Malley
October 1992 **ACM Transactions on Software Engineering and Methodology (TOSEM)**, Volume 1 Issue 4
Publisher: ACM Press
Full text available: pdf(3.15 MB) Additional Information: full citation, abstract, references, citings, index terms, review

We present a domain-independent model of hierarchical software system design and construction that is based on interchangeable software components and large-scale reuse. The model unifies the conceptualizations of two independent projects, Genesis and Avoca, that are successful examples of software component/building-block technologies and domain modeling. Building-block technologies exploit large-scale reuse, rely on open architecture software, and elevate the granularity of programming to ...

Keywords: domain modeling, open system architectures, reuse, software building-blocks, software design

11 Special issue on spatial database systems: An introduction to spatial database systems
Ralf Hartmut Güting
October 1994 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 3 Issue 4
Publisher: Springer-Verlag New York, Inc.
Full text available: pdf(2.50 MB) Additional Information: full citation, abstract, references, citings

We propose a definition of a spatial database system as a database system that offers spatial data types in its data model and query language, and supports spatial data types in its implementation, providing at least spatial indexing and spatial join methods. Spatial database systems offer the underlying database technology for geographic information systems and other applications. We survey data modeling, querying, data structures and algorithms, and system architecture for such systems. The em ...

12 The architecture of the EXODUS extensible DBMS

Michael J. Carey, David J. DeWitt, Daniel Frank, M. Muralikrishna, Goetz Graefe, Joel E. Richardson, Eugene J. Shekita

September 1986 **Proceedings on the 1986 international workshop on Object-oriented database systems**

Publisher: IEEE Computer Society Press

Full text available: pdf(1.68 MB) Additional Information: full citation, abstract, references, citings, index terms

With non-traditional application areas such as engineering design, image/voice data management, scientific/statistical applications, and artificial intelligence systems all clamoring for ways to store and efficiently process larger and larger volumes of data, it is clear that traditional database technology has been pushed to its limits. It also seems clear that no single database system will be capable of simultaneously meeting the functionality and performance requirements of such a diver ...

13 The FINITE STRING Newsletter: Abstracts of current literature

Computational Linguistics Staff

January 1987 **Computational Linguistics**, Volume 13 Issue 1-2

Publisher: MIT Press

Full text available: pdf(6.15 MB) Publisher Site Additional Information: full citation

14 Special issue on prototypes of deductive database systems: The aditi deductive database system

Jayen Vaghani, Kotagiri Ramamohanarao, David B. Kemp, Zoltan Somogyi, Peter J. Stuckey, Tim S. Leask, James Harland

April 1994 **The VLDB Journal — The International Journal on Very Large Data Bases**, Volume 3 Issue 2

Publisher: Springer-Verlag New York, Inc.

Full text available: pdf(2.67 MB) Additional Information: full citation, abstract, references, citings

Deductive databases generalize relational databases by providing support for recursive views and non-atomic data. Aditi is a deductive system based on the client-server model; it is inherently multi-user and capable of exploiting parallelism on shared-memory multiprocessors. The back-end uses relational technology for efficiency in the management of disk-based data and uses optimization algorithms especially developed for the bottom-up evaluation of logical queries involving recursion. The front ...

Keywords: implementation, logic, multi-user, parallelism, relational database

15 Special issue: AI in engineering

D. Sriram, R. Joobbani

April 1985 **ACM SIGART Bulletin**, Issue 92

Publisher: ACM Press

Full text available: pdf(8.79 MB) Additional Information: full citation, abstract

The papers in this special issue were compiled from responses to the announcement in the July 1984 issue of the SIGART newsletter and notices posted over the ARPAnet. The interest being shown in this area is reflected in the sixty papers received from over six countries. About half the papers were received over the computer network.

16 Potpourri: Alternative representations and abstractions for moving sensors databases

J. Eisenstein, S. Ghandeharizadeh, C. Shahabi, G. Shanbhag, R. Zimmermann

October 2001 **Proceedings of the tenth international conference on Information and knowledge management**

Publisher: ACM Press

Full text available: pdf(1.50 MB) Additional Information: full citation, abstract, references, index terms

Moving sensors refers to an emerging class of data intensive applications that inpacts disciplines such as communication, health-care, scientific applications, etc. These applications consist of a fixed number of sensors that move and produce streams of data as a function of time. They may require the system to match these streams against stored streams to retrieve relevant data (patterns). With communication, for example, a speaking impaired individual might utilize a haptic glove that translat ...

17 A structural view of the Cedar programming environment

Daniel C. Swinehart, Polle T. Zellweger, Richard J. Beach, Robert B. Hagmann

August 1986 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 8 Issue 4

Publisher: ACM Press

Full text available: pdf(6.32 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper presents an overview of the Cedar programming environment, focusing on its overall structure—that is, the major components of Cedar and the way they are organized. Cedar supports the development of programs written in a single programming language, also called Cedar. Its primary purpose is to increase the productivity of programmers whose activities include experimental programming and the development of prototype software systems for a high-performance personal computer. T ...

18 The model, language, and implementation of an object-oriented multimedia knowledge base management system

Hiroshi Ishikawa, Fumio Suzuki, Fumihiko Kozakura, Akifumi Makinouchi, Mika Miyagishima, Yoshio Izumida, Masaaki Aoshima, Yasuo Yamane

March 1993 **ACM Transactions on Database Systems (TODS)**, Volume 18 Issue 1

Publisher: ACM Press

Full text available: pdf(3.23 MB) Additional Information: full citation, abstract, references, citings, index terms, review

New applications such as CAD, AI, and hypermedia require direct representation and flexible use of complex objects, behavioral knowledge, and multimedia data. To this end, we have devised a knowledge base management system called Jasmine. An object-oriented approach in a programming language also seems promising for use in Jasmine. Jasmine extends the current object-oriented approach and provides the following features. Our object model is based on functional data models and well-establis ...

19 GPGPU: general purpose computation on graphics hardware

David Luebke, Mark Harris, Jens Krüger, Tim Purcell, Naga Govindaraju, Ian Buck, Cliff Woolley, Aaron Lefohn

August 2004 **Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04**

Publisher: ACM Press

Full text available: pdf(63.03 MB) Additional Information: full citation, abstract

The graphics processor (GPU) on today's commodity video cards has evolved into an extremely powerful and flexible processor. The latest graphics architectures provide tremendous memory bandwidth and computational horsepower, with fully programmable vertex and pixel processing units that support vector operations up to full IEEE floating point precision. High level languages have emerged for graphics hardware, making this computational power accessible. Architecturally, GPUs are highly parallel s ...

20 The family of concurrent logic programming languages
Ehud Shapiro
September 1989 **ACM Computing Surveys (CSUR)**, Volume 21 Issue 3
Publisher: ACM Press

Full text available: pdf(9.62 MB) Additional Information: full citation, abstract, references, citings, index terms

Concurrent logic languages are high-level programming languages for parallel and distributed systems that offer a wide range of both known and novel concurrent programming techniques. Being logic programming languages, they preserve many advantages of the abstract logic programming model, including the logical reading of programs and computations, the convenience of representing data structures with logical terms and manipulating them using unification, and the amenability to metaprogrammin ...

Results 1 - 20 of 200 Result page: **1** 2 3 4 5 6 7 8 9 10 next